PCT / IB 0 3 / 0 2 2 3 6
31. 07. 03
REC'D 05 AUG 2003
WIPO PCT

REGISTRY OF PATENTS
SINGAPORE

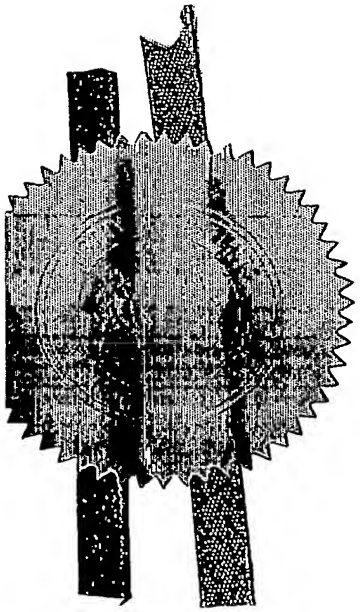
This is to certify that the annexed is a true copy of the following international application as filed with the Registry as the receiving Office and of any corrections thereto.

Date of Filing	:	24 SEP 2002
Application Number	:	PCT/SG02/00239 [withdrawn]
Applicant(s) / Proprietor(s) of Patent	:	PHILIPS ELECTRONICS SINGAPORE PTE LTD; KONINKLIJKE PHILIPS ELECTRONICS N.V.
Title of Invention	:	BUS STATION AND BUS SYSTEM

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

Tan Kar Leng (Ms)
Assistant Registrar
for REGISTRAR OF PATENTS
SINGAPORE

2 Jul 2003



HOME COPY

PCT REQUEST

PSG020018WOP

Original (for SUBMISSION) - printed on 24.09.2002 11:03:44 AM

0	**For receiving Office use only**	
0-1	International Application No.	PCT/SG 0 2 / 0 0 2 3 9
0-2	International Filing Date	2 4 SEP 2002 (24-09-02)
0-3	Name of receiving Office and "PCT International Application"	REGISTRY OF PATENTS (SINGAPORE) PCT INTERNATIONAL APPLICATION

0-4	**Form - PCT/RO/101 PCT Request**	
0-4-1	Prepared using	PCT-EASY Version 2.92 (updated 01.06.2002)
0-5	**Petition** The undersigned requests that the present international application be processed according to the Patent Cooperation Treaty	
0-6	**Receiving Office (specified by the applicant)**	Intellectual Property Office of Singapore (RO/SG)
0-7	**Applicant's or agent's file reference**	PSG020018WOP
I	**Title of invention**	BUS STATION AND BUS SYSTEM
II	**Applicant**	
II-1	This person is:	applicant only
II-2	Applicant for	SG
II-4	Name	PHILIPS ELECTRONICS SINGAPORE PTE. LTD.
II-5	Address:	Attn. Mr. J.C. De Visser 620A Lorong 1 Toa Payoh TP2-2nd Floor 319762 Singapore Singapore
II-6	State of nationality	SG
II-7	State of residence	SG
II-8	Telephone No.	+65 7995021
II-9	Facsimile No.	+65 7995022

CONFIRMATION COPY

PCT REQUEST

Original (for SUBMISSION) - printed on 24.09.2002 11:03:44 AM

III-1	Applicant and/or inventor	
III-1-1	This person is:	applicant only
III-1-2	Applicant for	all designated States
III-1-4	Name	KONINKLIJKE PHILIPS ELECTRONICS N.V.
III-1-5	Address:	Groenewoudseweg 1 NL-5621 BA Eindhoven Netherlands
III-1-6	State of nationality	NL
III-1-7	State of residence	NL
III-1-8	Telephone No.	+31 40 2743444
III-1-9	Facsimile No.	+31 40 2743489
III-1-11	Applicant's registration No. with the Office	GPA 02/0007
V	Designation of States	
V-1	Regional Patent (other kinds of protection or treatment, if any, are specified between parentheses after the designation(s) concerned)	EP: AT BE CH&LI CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE TR and any other State which is a Contracting State of the European Patent Convention and of the PCT (except BG CZ EE SK)
V-2	National Patent (other kinds of protection or treatment, if any, are specified between parentheses after the designation(s) concerned)	SG
V-5	Precautionary Designation Statement In addition to the designations made under items V-1, V-2 and V-3, the applicant also makes under Rule 4.9(b) all designations which would be permitted under the PCT except any designation(s) of the State(s) indicated under item V-6 below. The applicant declares that those additional designations are subject to confirmation and that any designation which is not confirmed before the expiration of 15 months from the priority date is to be regarded as withdrawn by the applicant at the expiration of that time limit.	
V-6	Exclusion(s) from precautionary designations	NONE
VI	Priority claim	NONE
VII-1	International Searching Authority Chosen	European Patent Office (EPO) (ISA/EP)

PCT REQUEST

PSG020018WOP

Original (for SUBMISSION) - printed on 24.09.2002 11:03:44 AM

VIII	Declarations	Number of declarations	
VIII-1	Declaration as to the identity of the inventor	-	
VIII-2	Declaration as to the applicant's entitlement, as at the international filing date, to apply for and be granted a patent	-	
VIII-3	Declaration as to the applicant's entitlement, as at the international filing date, to claim the priority of the earlier application	-	
VIII-4	Declaration of inventorship (only for the purposes of the designation of the United States of America)	-	
VIII-5	Declaration as to non-prejudicial disclosures or exceptions to lack of novelty	-	
IX	Check list	number of sheets	electronic file(s) attached
IX-1	Request (including declaration sheets)	4	-
IX-2	Description	13	-
IX-3	Claims	2	-
IX-4	Abstract	1	EZABST00.TXT
IX-5	Drawings	4	-
IX-7	TOTAL	24	
	Accompanying items	paper document(s) attached	electronic file(s) attached
IX-8	Fee calculation sheet	✓	-
IX-17	PCT-EASY diskette	-	Diskette
IX-19	Figure of the drawings which should accompany the abstract	2	
IX-20	Language of filing of the international application	English	
X	Signature of applicant, agent or common representative		
X-1	Name (LAST, First)	VAN DER VEER, Johannis, L.(Authorized Representative)	
X-2	Capacity		

FOR RECEIVING OFFICE USE ONLY

10-1	Date of actual receipt of the purported international application	24 SEP 2002 (24-09-02)
10-2	Drawings:	
10-2-1	Received	
10-2-2	Not received	
10-3	Corrected date of actual receipt due to later but timely received papers or drawings completing the purported international application	
10-4	Date of timely receipt of the required corrections under PCT Article 11(2)	
10-5	International Searching Authority	ISA/EP
10-6	Transmittal of search copy delayed until search fee is paid	

PCT REQUEST

PSG020018WOP

Original (for SUBMISSION) - printed on 24.09.2002 11:03:44 AM

FOR INTERNATIONAL BUREAU USE ONLY

11-1	Date of receipt of the record copy by the International Bureau	

Bus station and bus system

The invention relates to a bus station for use in a bus communication system, comprising a first communication port and a second communication port.

The invention further relates to a bus station for use in a bus system, comprising a device controller coupled to a communication port, being arranged to operate as a device station, said bus station further being arranged to operate under control of system software.

The invention further relates to a bus communication system comprising a first bus station comprising a device communication port coupled to a first communication port on a second bus station, said second bus station further comprising a second communication port.

SoftHost using VirtualHAL Technology

A New Architecture for Implementation of USB Host Dongle

18/9/2002

Ng Chee Yu, Alan Chang, Jerome Tjia, Kawshol Sharma
Philips Electronics Singapore

1 Background

In the recent years, USB has become a standard part of all PCs. Numerous computer peripherals are equipped with USB for connection to PCs. However, USB remains a PC-centric protocol where the PC is always the master (USB Host) of all other devices such as scanner, PDA, Mobile Phones, Printers and Harddisk are peripherals (USB Device). In a USB system, there can only be 1 host. The host controls up to 128 USB devices. USB devices can never communicate with each other without a host due to the architecture of USB.

Giving the ability of becoming a USB Host to certain devices such as PDA and Mobile Phones will open up a whole new arena of possibilities. PDA with USB Host capability may access a USB harddisk, or connects to the Internet using a USB modem. Mobile Phones with USB Host capability may access a ThumbDrive for SIM Card back up, etc. SoftHost is designed to provide USB Host capability to any embedded system that has
-An existing USB device hardware
-Adequate CPU/Memory resources

Together with VirtualHAL, SoftHost would allow USB device to take on the role of a USB Host and gain access to the wide range of functions offered by other existing USB devices.

Advantage of the SoftHost solution:
-The host dongle does not need to handle USB software, allowing very simple firmware/MCU (Low Cost)
-Host software is handled by the embedded system.

Terms that are specific to this project:
VirtualHAL – Mapping the hardware access of a target through a USB data link, to allow a system to access a remotely located hardware.

SoftHost – An application of VirtualHAL which allows a USB Device to attain the capability of a USB Host.

2 Objective

This project aims to provide an easy and low cost upgrade path to USB equipped embedded systems. To achieve this, a hardware dongle and a driver update will be required. The hardware dongle consists of an Embedded USB Host Controller and a low-end MCU. This dongle is designed to work with any USB equipped embedded systems without modification. The driver update is system specific, and it serves to add in the VirtualHAL to the existing USB hardware.

3 VirtualHAL

VirtualHAL (HAL stands for Hardware Abstraction Layer) is a combination of 3 components:

- Software layer
- Existing USB-Device hardware
- USB Host and an MCU running VirtualHAL firmware

VirtualHAL provides complete access to the target hardware on the connected dongle, using the USB-Device hardware. In other words, ***the existing USB-Device hardware is used as an asynchronous microprocessor interface bus*** to allow the USB Host Driver to access the target hardware.

One possible application of VirtualHAL is SoftHost. SoftHost is a hardware dongle that allows a system with USB Device to attain the capability of USB hardware without changes to the existing hardware (see Figure. 1).

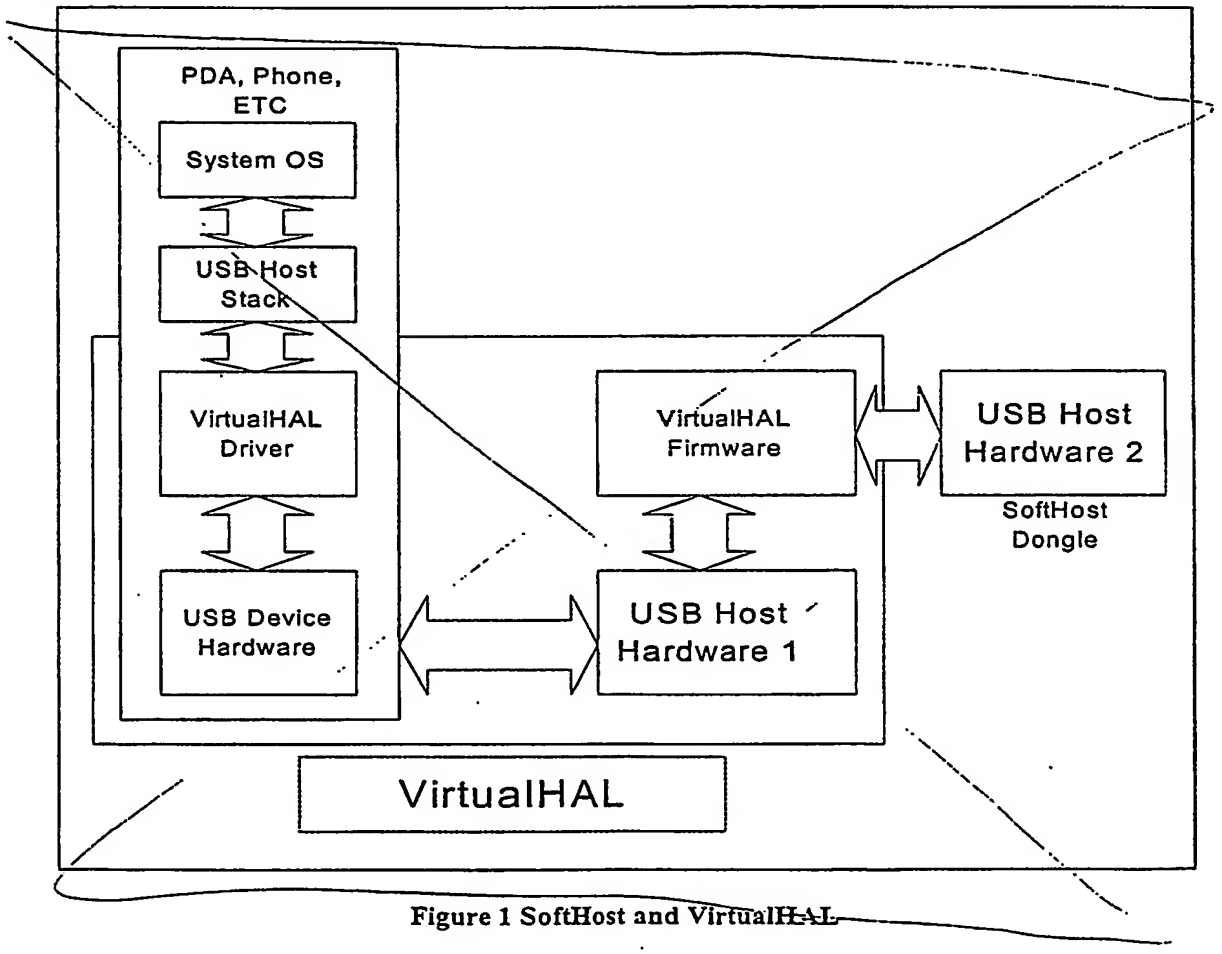


Figure 1 SoftHost and VirtualHAL

VirtualHAL could be used to add in multiple interface/functions to a system with USB Device capability, for example: Bluetooth, IrDA, USB-OTG, etc.

4 Application : SoftHost

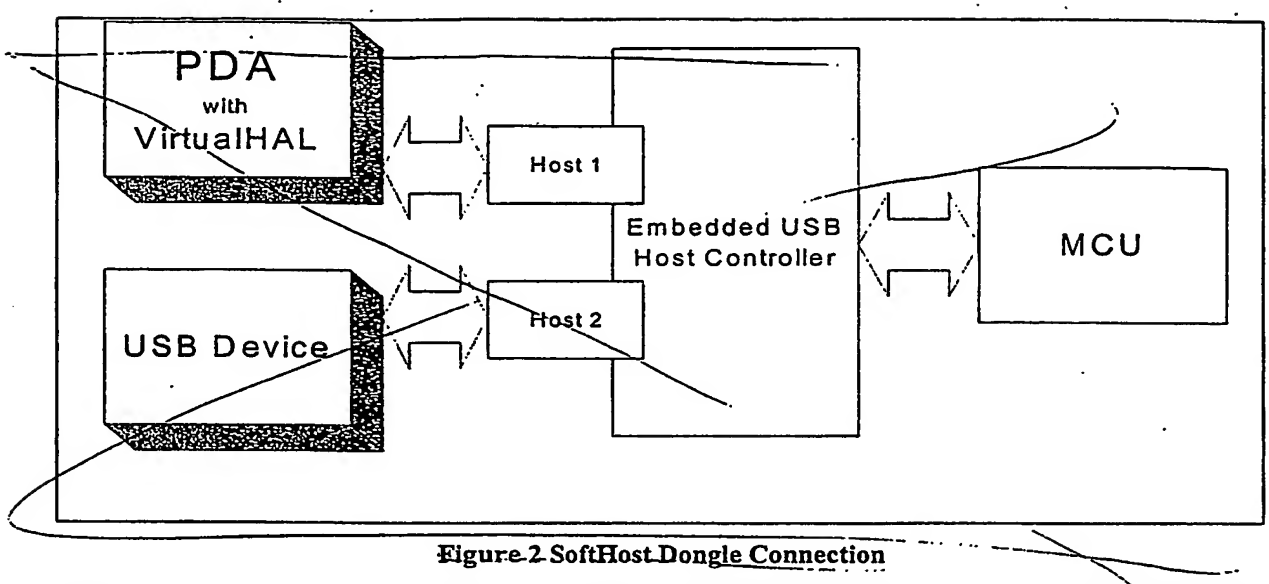


Figure 2 SoftHost Dongle Connection

SoftHost dongle is made up of 2 main components: 1 or 2 embedded USB Host Controller, which would be referred as Host 1 and Host 2 and a low-power MCU (see Figure 2). Host 1 of the SoftHost Dongle will be connected to the USB Device Port of the PDA. Host 2 of the dongle will function as a Host Port to the PDA. Thus, the PDA may make use of a mouse connected to Host 2, or print directly to a printer at Host 2, despite of the fact that there is no built-in USB Host hardware in the PDA!

Note that Host 1 and Host 2 could be 2 host ports of the same USB Host Hardware, or 2 independent USB Host Hardware.

The MCU has a number of tasks to execute:
Upon powering up, it polls the H1 for connection of any USB Device. Upon detecting a connection, it enumerates the device and checks if it is a device running VirtualHAL. If the device is indeed running VirtualHAL, the MCU will go into operational mode. Otherwise, it will disable the device and trigger a flashing LED, signaling that connected device does not support VirtualHAL.

In operational mode, the MCU sets up an interrupt pipe and polls the VirtualHAL for SoftHostPacket every millisecond. If the Host Stack on the PDA has send a hardware access request through the VirtualHAL, the VirtualHAL will send the request as a SoftHostPacket when the H1 polls it though the interrupt pipe. The MCU will retrieve this SoftHostPacket from the Embedded USB Host Controller buffer memory and execute the hardware access accordingly. If there are data to be returned (read operations), the MCU will send out the corresponding data through the H1.

The following figure compares 3 systems: PC with Host only, PDA with Device only, and a PDA with Device only connected to a SoftHost Dongle (see Figure 3).

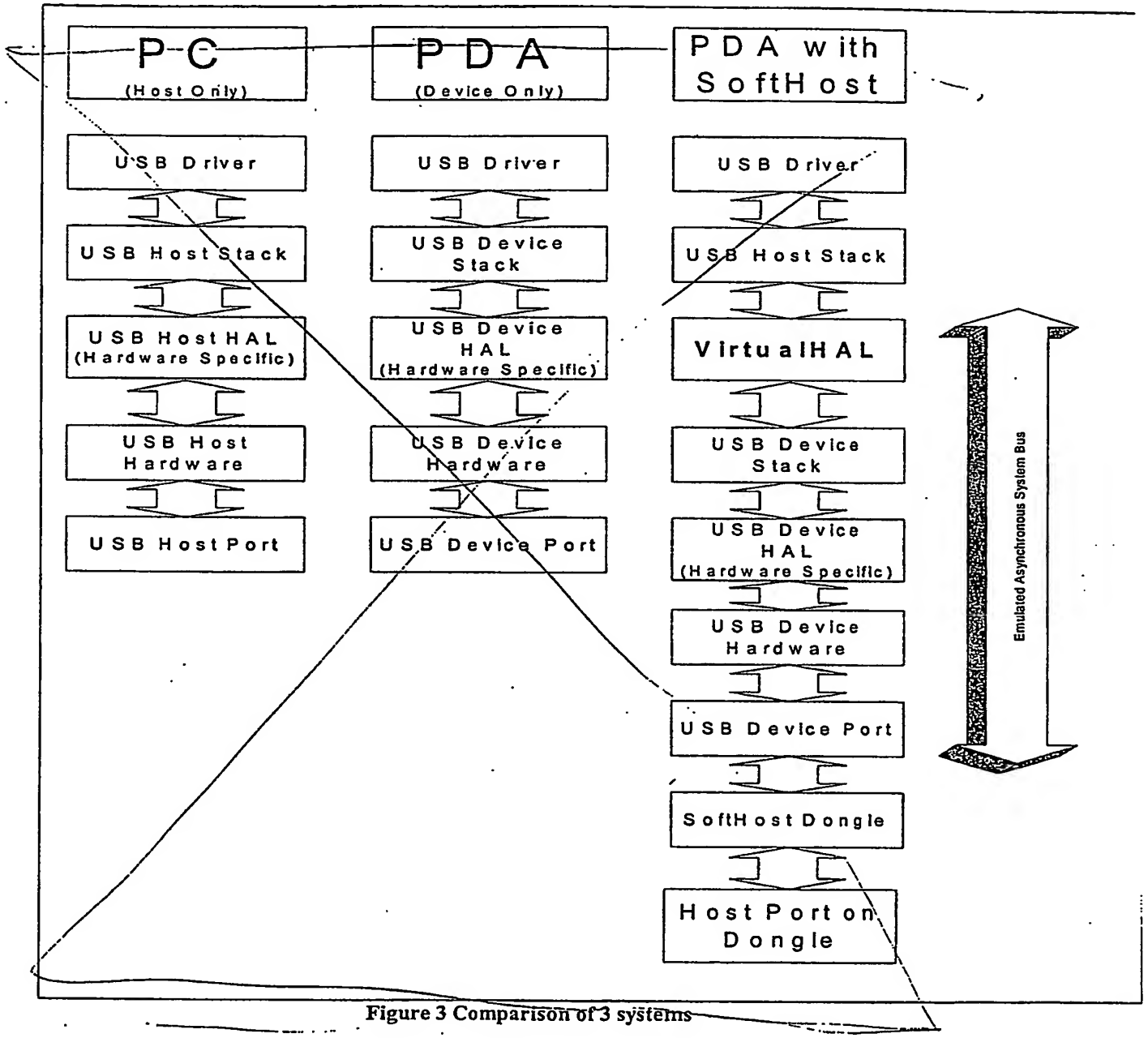


Figure 3 Comparison of 3 systems

In the PC with Host system, the Host Stack accesses the underlying USB Hardware through the Host HAL. Similarly, the PDA with device, Device stack accesses the underlying USB hardware though the Device HAL.

In the SoftHost system, when the Host Stack needs to access the Host Hardware, it communicates the access operation details to the VirtualHAL. The VirtualHAL wraps these details in a pre-determined format

– SoftHost Protocol. The SoftHost protocol packet is sent out through the existing USB Device Hardware when the SoftHost Dongle polls it for outstanding SoftHost Protocol packets.

SoftHost Protocol provides the following access functions:
-Reading a register in Dongle Host Hardware
-Writing a register in Dongle Host Hardware
-Reading buffer memory in Dongle Host Hardware
-Writing buffer memory in Dongle Host Hardware

More advances functions would be added to improve the system performance, for example, a function that reads a register, AND/OR it with a value and writes back into the register.

To the Host Stack, there would be no difference between communicating with a Host HAL, or in this case, the VirtualHAL. Host Stack will see an actual Host Hardware through the VirtualHAL.

5 SoftHost Protocol

This protocol defines the method on which, Host Stack running on VirtualHAL may access the hardware of the Host Controller using the Device Controller hardware. Figure 4 illustrates the location of the SoftHost Protocol layer in a USB system.

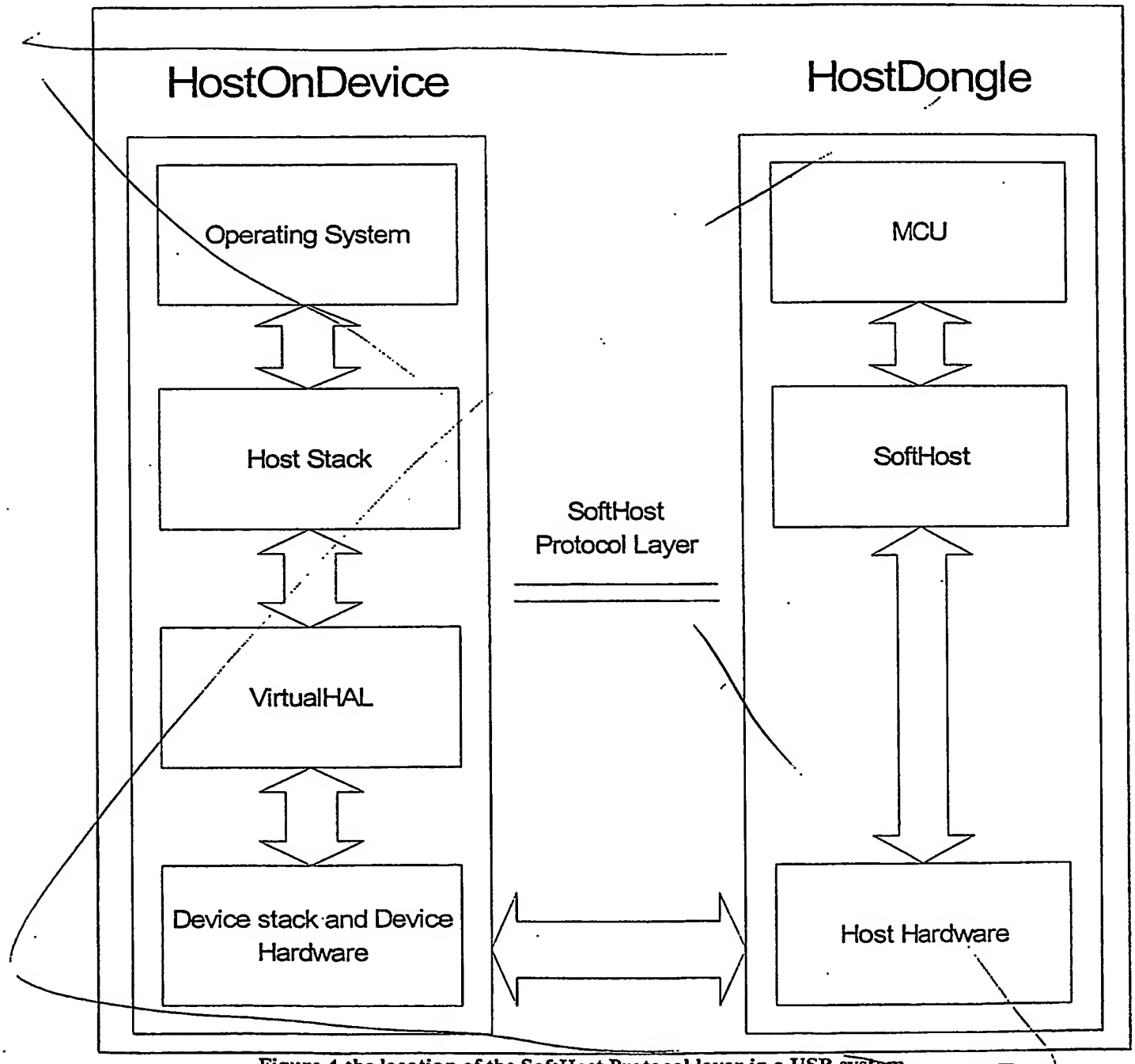


Figure 4 the location of the SoftHost Protocol layer in a USB system.

The SoftHost protocol starts at the moment where:
-HostDongle has enumerated the connected device
-The connected device is confirmed to be a HostOnDevice

Special terms used:
HostDongle : A USB Host coupled to an MCU running SoftHost firmware.
HostOnDevice : An embedded system with USB device hardware, running Host Stack on VirtualHAL.

5.1 The Traffic

The HostDongle and the HostOnDevice communicates using a dedicated bi-directional bulk pipe. There are 4 types of payloads:

5.1.1 HRU (HostDongle Request Unit)

- Sent by HostDongle
- A bulk packet of 8 bytes payload
- Used for polling the HostOnDevice
- May contain Interrupt Information. (HRU_IRQ)
- HostDongle ALWAYS send a bulk-in of 1023 bytes after sending the HRU. HostOnDevice would reply with NOB or CRP through this bulk-in

5.1.2 NOB (No Outstanding Business)

- Sent by HostOnDevice
- A bulk packet of 8 bytes payload
- Sent when there are no outstanding transactions

5.1.3 CRP (Command Request Packet)

Sent by HostOnDevice
A bulk packet of 16-1023 bytes
Contains a number of command sets, and an optional data set

5.1.4 APR (As Per Requested)

Sent by HostDongle
A bulk packet of 16-1023 bytes
Contains the results of previously received CRP commands, and an optional data set

5.2 The Flow

As in all USB system, all transfers start with an action by the Host. In the case of SoftHost protocol, SoftHost Dongle is always the Host. All SoftHost transfer cycles starts with a HRU. The current transfer cycle must be completed before the HostDongle starts on the next transfer cycle.

5.2.1 Poll-Nothing Cycle : HRU-NOB

HostDongle polls the HostOnDevice for outstanding command sets. There is no outstanding command sets, HostOnDevice replies with NOB.

Transactions:
HostDongle sends BULK-OUT
HostDongle sends DATA (HRU)
HostOnDevice sends ACK

HostDongle sends BULK-IN
HostOnDevice send DATA (NOB)
HostDongle sends ACK

5.2.2 Poll-Something Cycle : HRU-CRP-APR

HostDongle polls the HostOnDevice for outstanding command sets. HostOnDevice sends outstanding command set by CRP. HostDongle executes the command and returns the results by APR.

Transactions:
HostDongle sends BULK-OUT
HostDongle sends DATA (HRU)
HostOnDevice sends ACK

HostDongle sends BULK-IN
HostOnDevice send DATA (CRP)
HostDongle sends ACK

HostDongle sends BULK-OUT
HostDongle sends DATA (APR)
HostOnDevice sends ACK

5.2.3 Interrupt Cycle : HRU_IRQ-CRP-APR

HostDongle alerts the HostOnDevice on outstanding hardware interrupts. HostOnDevice decides on the appropriate command sets and sends it by CRP. HostDongle executes the commands and returns the results by APR. HostOnDevice MUST clear the outstanding interrupt or disable the generation of HRU_IRQ, or the HostDongle would sends HRU_IRQ continuously.

Transactions:
HostDongle sends BULK-OUT
HostDongle sends DATA (HRU_IRQ)
HostOnDevice sends ACK

HostDongle sends BULK-IN
HostOnDevice send DATA (CRP)
HostDongle sends ACK

HostDongle sends BULK-OUT
HostDongle sends DATA (APR)

HostOnDevice sends ACK

5.3 Packet Formats

5.3.1 HRU Format

The HRU contains the following information:
-Current Frame Number
-HcInterruptStatus of HC in HostDongle
–Interrupt Status of DC

5.3.2 NOB Format

-No special information required

5.3.3 CRP and APR Format

Active bit in Header is 1 for CRP, and 0 for APR.

CRP can be of a size of 16 – 1023 bytes. The total size is made up of
A number of command sets (8 bytes each)
An optional data set.

Maximum number of command sets in a CRP is 8. Maximum size of data set is 1023-(8*number of command sets)

The multiple command sets in a single command request packet allows a sequence of hardware accesses to be communicated in a single packet thus reduces the latency of transfer.

Command Set Format

Command Set is an 8-byte data structure. It contains the following information:
-Command Set Header (1 byte)
-Command Set Index (2 bytes)
-Command Set Data (4 bytes)
-Command Set Aux (1 byte)

Bit	7	6	5	4	3	2	1	0
Group	Active	Remaining Sets			OpCode			
Attribute	Boolean	0-7. 0 means last set			0-15			

OpCode	Operation by MCU
0	Write [Aux] bytes from [Data] into [Index] register
1	Read [Aux] bytes from [Index] register into [Data]
2	Write [Data] bytes from DataSet into address location [Index]
3	Read [Data] bytes into DataSet from address location [Index]
4	Read [Aux] bytes from [Index] register, OR with [Data] and write back into [Index] register
5	Read [Aux] bytes from [Index] register, AND with [Data] and write back into [Index] register
6	Set polling rate to [Index]
7	Set HRU_IRQ On/Off
8-15	Reserved. No Action by MCU

6 Adaptation of SoftHost Protocol for other applications

SoftHost Protocol is defined to be a generic protocol, and could be easily adapted for other applications. For example, a Bluetooth adapter for a system with existing USB Device Hardware: SoftBluetooth. VirtualHAL Driver and firmware requires minimal or no changes (see Figure 5).

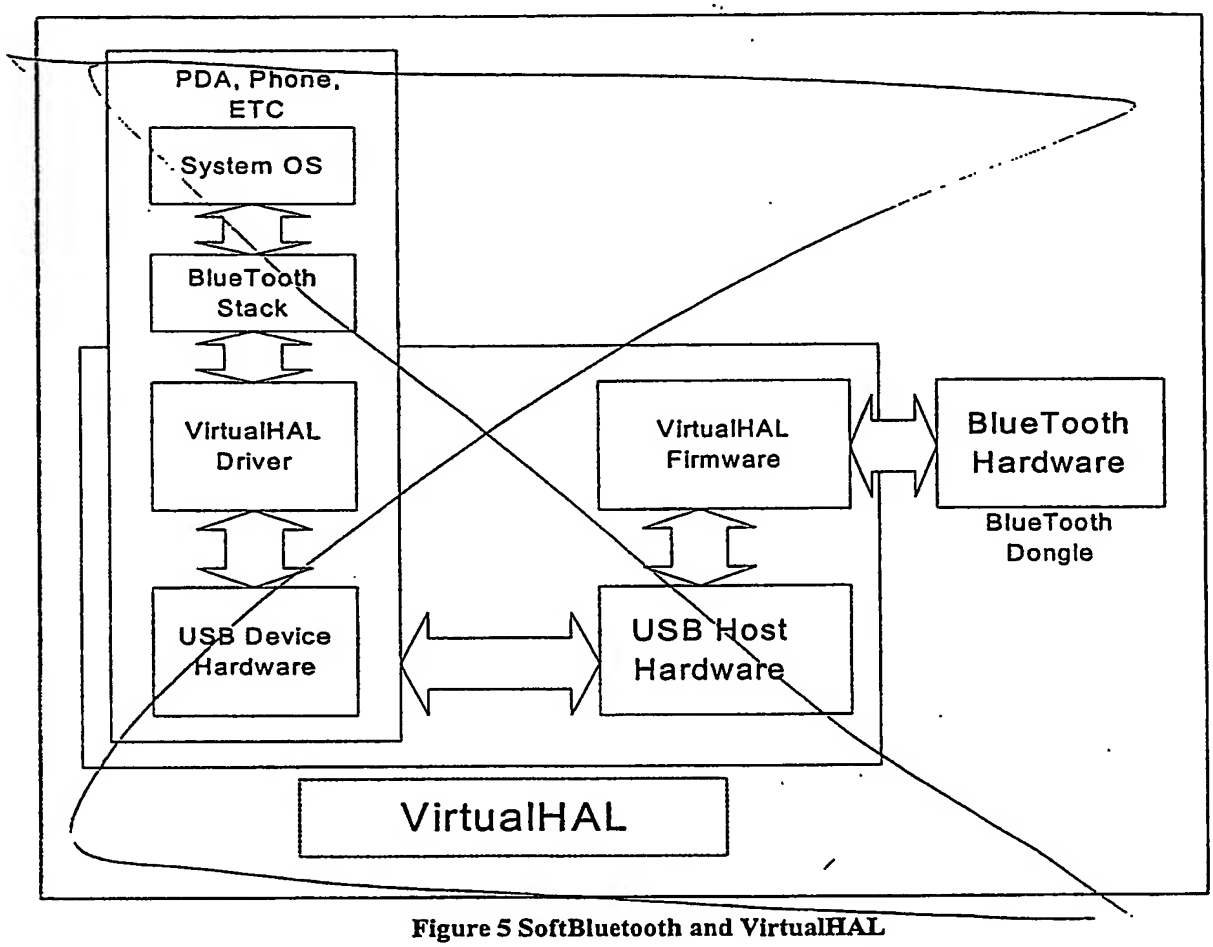


Figure 5 SoftBluetooth and VirtualHAL

7 CLAIMS

1. A bus station for use in a bus communication system, comprising a first communication port and a second communication port, being arranged to operate in a first mode upon detection of the presence of a host station coupled to said second port and to operate in a second mode upon detection of the absence of a host station coupled to said second port, said bus station being arranged in said first mode of operation to pass communication between said host station coupled to said second port and a device station coupled to said first port, said bus station further being arranged to operate as an alternate host station in said second mode of operation, by communicating with said device station coupled to said first port according to a communication protocol whereby said bus station initiates communications.

2. A bus station according to claim 1 wherein said bus station is arranged to operate as a USB hub in said first mode of operation and to operate as a USB host in said second mode of operation

3. A bus station according to claim 1 wherein said bus station further comprises hub circuitry coupled to said first and second port for passing communication between said host station coupled to said second port and said device station in said first mode of operation.

4. A bus station for use in a bus system, comprising a device controller coupled to a communication port, being arranged to operate as a device station, said bus station further being arranged to operate under control of system software, comprising an operating system and host station driver software being arranged to communicate with a host controller and to pass information to and from the operating system, wherein said system software further comprises host emulation software being arranged to emulate the presence of a host controller towards the host station driver software and the presence of device station driver software towards the device controller, further being arranged to translate

communications from the host station driver software to the device controller and vice versa.

5. A bus communication system comprising a first bus station comprising a device communication port coupled to a first communication port on a second bus station, said second bus station further comprising a second communication port, said second bus station being arranged to operate in a first mode upon detection of the presence of a host station coupled to said second port and to operate in a second mode upon detection of the absence of a host station coupled to said second port

6. A bus communication system according to claim 5, wherein said first station comprises a device controller coupled to said device communication port and being arranged to operate under control of system software, comprising an operating system and host station driver software being arranged to communicate with a host controller and to pass information to and from the operating system, wherein said system software further comprises host emulation software being arranged to emulate the presence of a host controller towards the host station driver software and the presence of device station driver software towards the device controller, further being arranged to translate communications from the host station driver software to the device controller and vice versa.

8 ABSTRACT

The invention relates to a bus system comprising a first bus station comprising a device communication port coupled to a first communication port on a second bus station, said second bus station further comprising a second communication port, said second bus station being arranged to operate in a first mode upon detection of the presence of a host station coupled to said second port and to operate in a second mode upon detection of the absence of a host station coupled to said second port

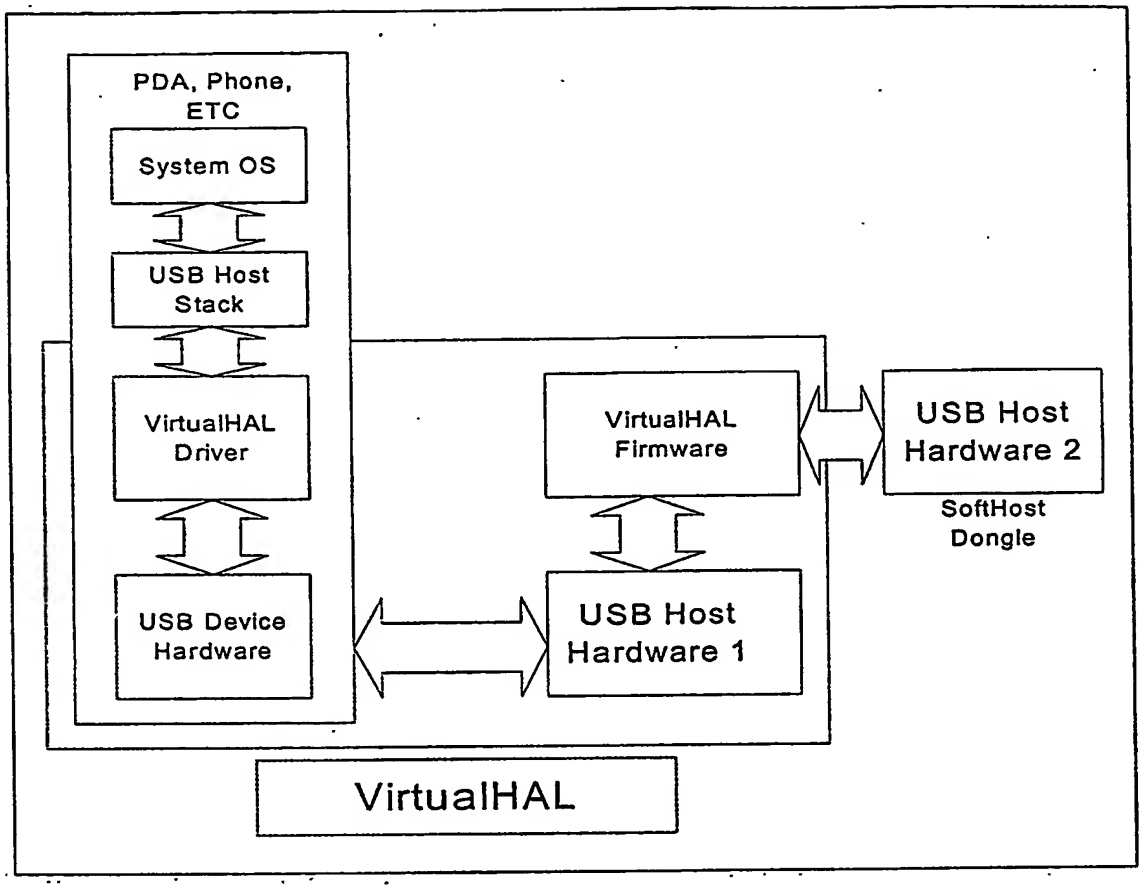


Figure 1 SoftHost and VirtualHAL

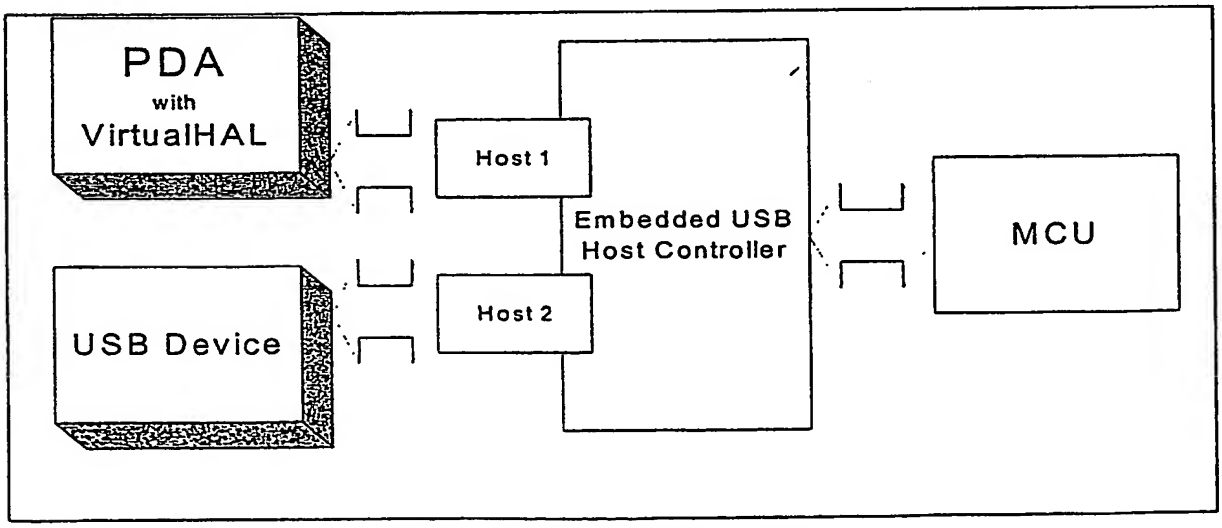


Figure 2 SoftHost Dongle Connection

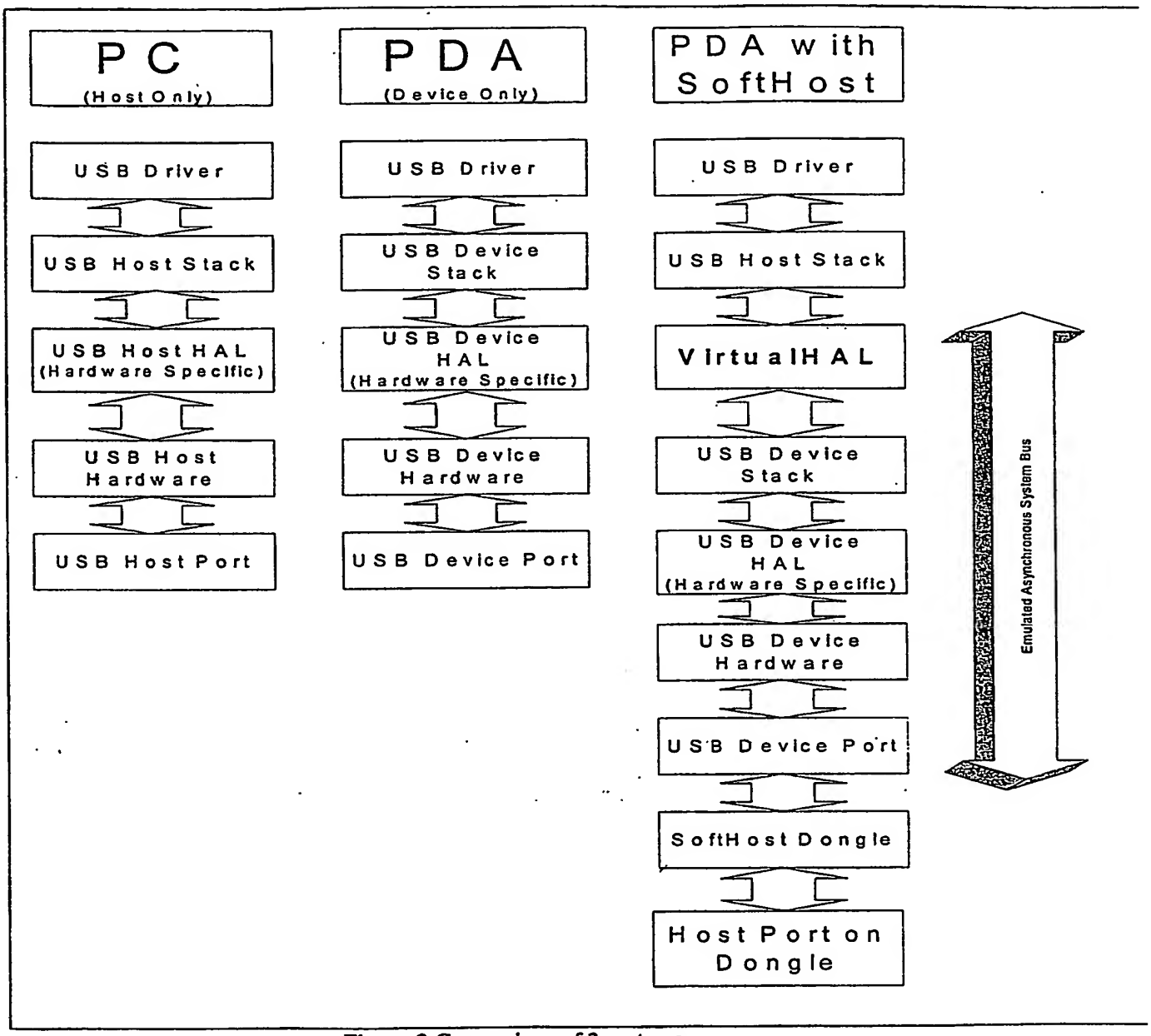


Figure 3 Comparison of 3 systems

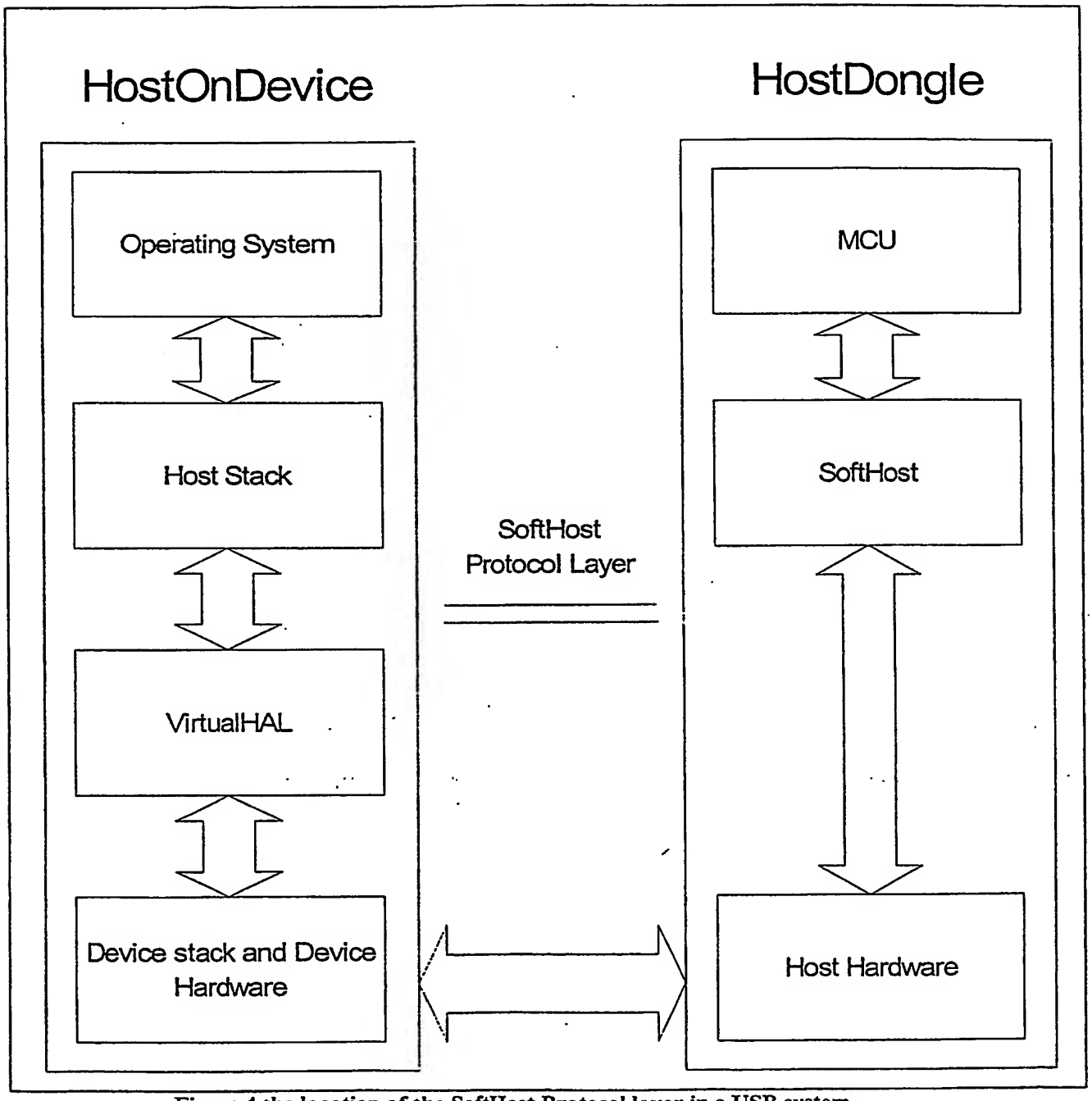


Figure 4 the location of the SoftHost Protocol layer in a USB system.

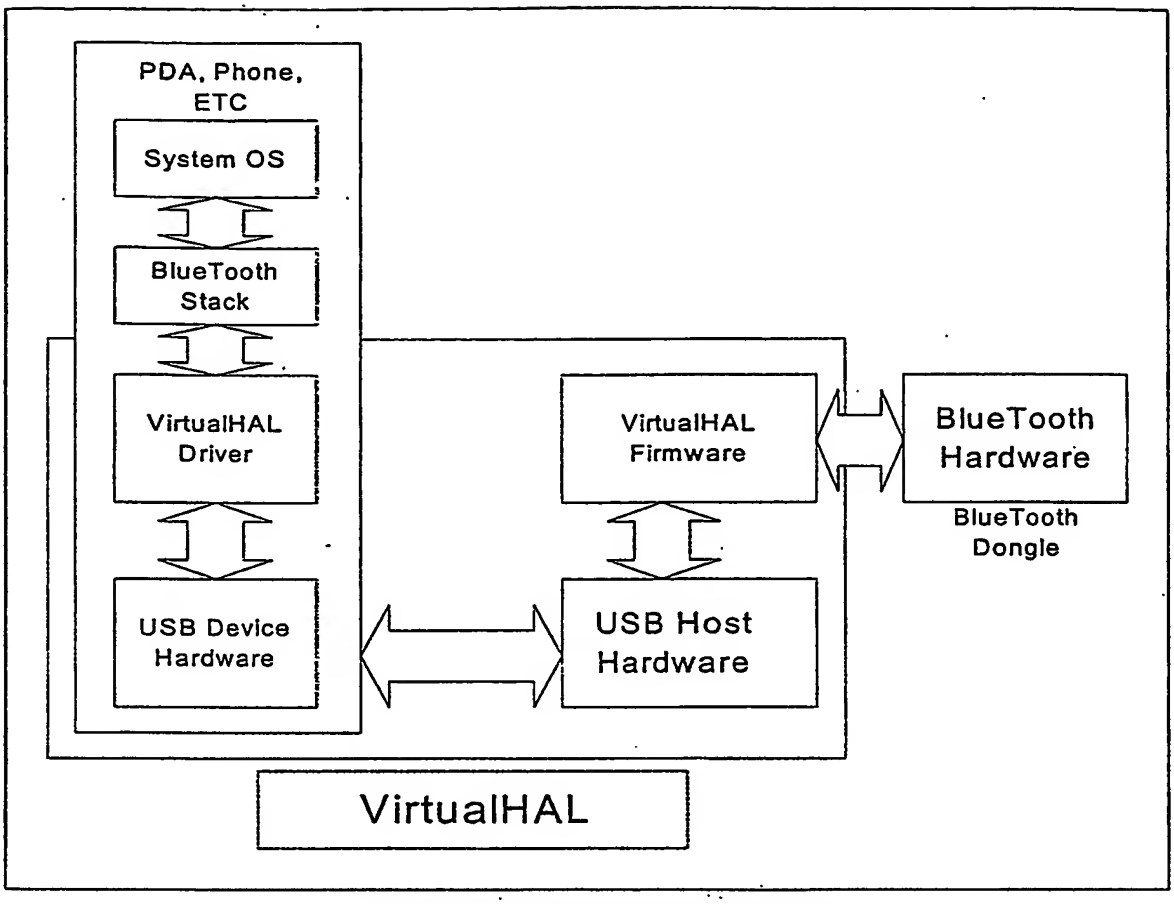


Figure 5 SoftBluetooth and VirtualHAL